GlobalTrust: An Attack-Resilient Reputation System for Tactical Networks

Xin Chen
Penn State University
University Park, PA 16802
xvc5038@cse.psu.edu

Jin-Hee Cho
U.S. Army Research Laboratory
Adelphi, MD 20783
jinhee.cho@us.army.mil

Sencun Zhu
Penn State University
University Park, PA 16802
szhu@cse.psu.edu

***Abstract*—In a military tactical network where a trust authority (e.g., a commander) makes a decision during a mission, assessing the trustworthiness of participating entities accurately is critical to mission success. In this work, we propose a trust-based reputation management scheme, called *GlobalTrust*, for minimizing false decisions on the reputation of nodes in the network. In the proposed scheme, nodes may be compromised and provide incorrect opinions to the trust authority, who conducts reputation evaluation towards all nodes based on the provided opinions. *GlobalTrust* achieves three goals: (1) maintaining a consistent global view towards each node; (2) obtaining high resiliency against various attack patterns; and (3) attaining highly accurate reputation values of nodes. Through extensive simulations comparing GlobalTrust with other existing schemes, we show that GlobalTrust minimizes false decisions while maintaining high resilience against various attack behaviors. Specifically, under various attacks, *GlobalTrust* can achieve a highly accurate consistent view on nodes' reputations even when the number of malicious nodes is up to 40% of all participating nodes.**

***Keywords*-Trust, Security, Reputation, Tactical networks.**

I. Introduction

Military tactical networks often face challenges in designing security protocols because they require additional precautions compared to civilian networks, including high hostility, distributed network characteristics, node subversion, and node heterogeneity. The mixture of wired/wireless communication mediums and high tempo operations cause rapid changes in network topology and service requirements. Since communities of interest (e.g., mission/task teams) are formed dynamically, participating nodes may not have any pre-defined trust relationships to each other. A tactical network may consist of heterogeneous entities characterized by humans (e.g., soldiers), robots, or unmanned/manned vehicles equipped with devices such as machines and/or sensors. In this work, we use the terms *a node* and *an entity* interchangeably to represent heterogeneous entities (or nodes) above. Military tactical networks typically have a hierarchical structure where a commander makes critical decisions to control all other entities in the network [1]. In this scenario, for the commander, it is critical to perceive an accurate view towards other entities for making right decisions. For example, when the commander wants to form a temporary mission team, called *a military coalition*, based on an acceptable trust level of nodes, the accuracy of trust assessment towards each node significantly impacts mission success.

One of the common applications using trust management mechanisms is to identify malicious entities in order to protect the network from attackers. The malicious entities may disrupt system security goals by performing network attacks such as loss of service availability (e.g., denial-of-service, packet dropping), and/or loss of data integrity (e.g., good/bad mouthing, message forgery/modification). In this work, we propose an attack-resilient reputation management mechanism that can accurately assess nodes' trustworthiness in the presence of highly hostile entities.

Trust or reputation management has been extensively studied in various domains [2]. In particular, many studies define *reputation* as a global perception of a node's trustworthiness in a network, whereas *trust* indicates an individual node's perception of any other node's trustworthiness based on its direct observation. A desirable reputation management should be able to provide the following features in the network:

- **Consistency**: provide a consistent view of the reputation of a node based on the consensus of honest nodes.
- **Resiliency**: be resilient to common security threats.
- **Accuracy**: derive valid reputation values based on accurate trust assessment.

Maintaining a consistent global view towards the node's reputation is challenging with uncertain or incomplete evidence in hostile, distributed tactical network environments.

This work proposes a reputation system, the so called *GlobalTrust*, for tactical networks for maximizing correct decision-making by identifying malicious entities. *GlobalTrust* has the following unique contributions: (1) it provides an accurate, consistent view on the reputation of all nodes and detects malicious nodes in the network; (2) it can effectively deal with various attack behaviors; and (3) it outperforms the existing reputation schemes (i.e., two schemes in *PeerTrust* [3]) in terms of view consistency and resilience against various attack behaviors.

II. Related Work

Trust or reputation management (TRM) schemes have been extensively studied in various domains. In the literature, the term *trust management* has been often interchangeably used with the term *reputation management* [4]. However, some researchers discussed the difference between trust and reputation. In [2], [5], trust is defined as a node's belief in trusting a peer, a subjective view towards its peer, while reputation

means the perception about a node formed by other peers. Thus, reputation can be estimated based on the aggregation of peer nodes' trust values.

Aberer and Despotovic [6] presented a trust-based reputation management scheme that is scalable for data management without any centralized control, but without considering collusive attacks. Kamvar *et al.* [7] proposed a distributed and secure method for reputation management that effectively identifies and isolates malicious nodes using the pre-trusted authority. Xiong and Liu [3] proposed two reputation-based trust models to evaluate a node's reputation in a fully distributed manner: *trust-value based credibility measure* (TVM) and *personalized similarity measure* (PSM). However, TVM is vulnerable to collusion attacks while PSM generates discrepancies in reputation about the same entity by different evaluators.

Zhou and Hwang [8] introduced a reputation system using power-law feedback provided by power nodes to aggregate reputation values in order to build a robust P2P system. Bella *et al.* [9] proposed a reputation management scheme that enables a node to exchange and update other nodes' reputation values in mobile ad hoc networks (MANETs). Arboit *et al.* [10] introduced a computational reputation model considering accusations against nodes in MANETs. However, [9], [10] do not deal with a false recommendation attack that often significantly deters accurate reputation assessment. Some other existing reputation management schemes [11]–[14] evaluate reputation of a node subjectively based on the evaluator's direct observation, ultimately leading to inconsistent global reputation view.

Quorum-based attack detection mechanisms has been extensively studied based on *k-out-of-n* threshold signatures [15], [16]. The key idea behind this is to determine the threshold $k+1$ as an upper bound of negative votes to diagnose a node as compromised. However, it is not trivial to obtain a sufficient number of votes under highly dynamic network environments. In addition, this work did not consider any collusive attack. Later, [17], [18] proposed mutual revocation based decision making schemes using the k-means clustering algorithm for trust management. The k-means-based judgment scheme, however, is vulnerable to a conflicting recommendation attack.

III. Preliminaries

A. Problem Statement and Challenges

We assume each node in a tactical network is pre-installed with a monitoring mechanism [19] characterized by detection error probability ε. This enables a node to directly observe its neighboring nodes' behavior. With this monitoring capability, each node can derive Local Trust Opinions (LTOs) about its neighboring nodes based on direct observations. For example, $LTO_{w,u}$ is node w's trust opinion towards node u based on *direct* observations. If node w has not encountered with node u, there will be no LTO. Let $p_{w,u}$ and $n_{w,u}$ be the total number of positive events and total number of negative events that node w observed about node u, over the period of encountering time. The LTO of node w towards node u during this time period, $LTO_{w,u}$, is calculated as:

$$LTO_{w,\,u} = \frac{p_{w,\,u}}{p_{w,\,u} + n_{w,\,u}} \tag{1}$$

$LTO_{w,u}$ is a real number scaled in [0, 1]. Note that if the total number of observed events, $p_{w,u} + n_{w,u}$, is 0 (i.e., no direct observation), $LTO_{w,u}$ will be set as a $null$ value. These LTOs form an LTO matrix where each entry $LTO_{i,j}$ is the LTO of node i towards node j. The following is a simple example of an LTO matrix with six nodes in the network, where the fourth and sixth nodes are malicious nodes giving false (dishonest) LTOs. Here empty entries indicate $null$ values.

$$\text{LTO} = \begin{pmatrix} & 0.82 & 1 & 0 & & 0.26 \\ 0.93 & & & & 0.88 & 0.20 \\ 0.96 & & & 0 & 0.93 & \\ 0.05 & 0 & 0 & & 0 & 1 \\ & & 0.89 & 0.18 & & 0.23 \\ 0 & 0 & 0.07 & 1 & 0 & \end{pmatrix} \tag{2}$$

We define the density of an LTO matrix, denoted as d, as the proportion of non-null LTOs (i.e., real values) in the matrix and calculate d as follows:

$$d = \frac{|\{(i,\,j) : LTO_{i,\,j} \neq null\}|}{N(N-1)} \tag{3}$$

where N is the number of nodes. Besides, every LTO is time-stamped to keep track of its freshness. Every node may store its LTOs using, for example, in-network storage technology with multiple copies to mitigate the potential data loss in a distributed network environment. That is, LTOs are stored and fetched in a distributed hash table (DHT) like P-Grid [6].

Given an LTO matrix for a given time period, our goal is to develop a reputation management scheme that can provide the network authority (e.g., a commander) with the capability of *consistent* and *accurate* assessment on the reputation of every node. That is, the proposed scheme aims to meet the following key requirements: (1) providing a consistent reputation value towards a node based on a LTO matrix; and (2) minimizing the inaccuracy of reputation evaluation introduced by intentionally injected false LTOs and imperfect monitoring error. To achieve these goals, we face two major challenges:

- **No pre-trusted LTOs**: The nodes which provide LTOs are not pre-trusted, so their LTOs cannot be trusted. In other words, the commander node cannot directly use these LTOs to derive reputation values for nodes.
- **Incomplete/Sparse LTO matrix**: The LTO matrix may be incomplete and even sparse due to the lack of observations or malicious nodes suppressing their LTO reports during an evaluation period.

B. Networw Model and Assumptions

GlobalTrust is a very generic framework, as long as it has an LTO matrix as the input. The LTO matrix can be generated from any group where members rate each other. Hence, *GlobalTrust* can be applied to the context of MANETs, peer-to-peer networks, Internet, or social networks. For concreteness, we assume that the targeted environment is a tactical network consisting of multiple mobile nodes communicating through

multiple hops. For secure communication, each node is pre-loaded with a public/private key pair or pairwise shared keys.

In our work, a network is allowed to be hierarchical in that nodes may have different ranks in the structure. Node k's hierarchical rank, HR_k, represents the importance of its role in the network. For instance, it is a very common scenario in a tactical network where entities with different ranks, such as a commander and his/her members, collaborate in a common mission.

In the considered military scenario, we allow a trusted authority (TA), such as a commander node, to be online periodically or as needed to collect evidence to assess reputation of other nodes and make trust decisions. None of the regular nodes is pre-trusted. We note that if in certain scenarios, when a single TA involves a security, safety, and/or performance concern, standard protocols [20] can be hold to distribute such a trust role into multiple regular nodes in the network, leading to reaching a consensus on the trustworthiness of all nodes. Such extensions are orthogonal to the reputation management algorithm in *GlobalTrust*.

A node may behave honestly, or may be compromised and perform various types of attacks. Now we describe various types of attack behaviors considered in this work below. We assume that honest nodes are a majority in the network, not allowing the Byzantine Failure condition due to too many malicious entities in the network. we demonstrate the impact of the ratio of malicious nodes on decision accuracy in Section V-C. As a measure of reputation, we consider the degree of compliance with a given network protocol (i.e., not performing network attacks and reporting honest LTOs compared against those of majority entities) by an entity.

C. Adversary Model

A malicious node (*aka* a compromised node or attacker) is defined as a node not complying with a given network protocol by either denying requested services or providing false LTOs. We model the degree of an attacker's misbehavior with attack intensity, α, ranged in $[0,\ 1]$. With this attack intensity, we can model a random attack behavior where an attacker performs an attack with probability α while exhibiting honest behavior with probability $(1-\alpha)$. Malicious nodes may collude to promote their reputations via good mouthing attacks while demoting honest nodes' reputations via bad mouthing attacks. Malicious nodes may provide false LTOs that are opposite to their actual observations. In this work, we consider the following attack behaviors:

- **Naïve Malicious Attack (NMA)**: A compromised node may provide improper services, not complying with a given network service protocol. However, it does not lie in reporting its LTOs.
- **Collusive Rumor Attack (CRA)**: In addition to providing improper services, malicious nodes collude to report false LTOs (i.e., good/bad mouthing attacks) for disrupting accurate trust or reputation assessment.
- **Non-collusive Rumor Attack (NRA)**: Without colluding with other malicious nodes, a malicious node can report a false LTO that is opposite to the observed evidence. For example, if an LTO is evaluated as p, the malicious node may report $1-p$ for the LTO.
- **Malicious Spy Attack (MSA)**: Some malicious nodes misbehave while other malicious nodes, called *malicious spies*, behave normally by providing proper services. These malicious nodes may collude and form a attacker community to perform good/bad mouthing attacks by reporting false LTOs, in order to subvert the entire trust and reputation system [21].
- **Conflicting Behavior Attack (CBA)**: Malicious nodes can behave inconsistently to different parties. This attack aims to disseminate conflicting (or inconsistent) LTOs. For example, they may misbehave only to a subset of honest nodes (referred to as *target nodes*) to intensify the LTO discrepancy between targeted and non-targeted honest nodes. This attack may reduce the overall attack intensity due to the nature of intermittent misbehavior.

IV. GlobalTrust

A. Overview

With a commander node taking the role of TA, *GlobalTrust* is deployed on TA to evaluate the global reputations of all nodes. Whenever TA comes online, it collects all LTOs with timestamps during the last offline interval. In this section, we discuss how to evaluate global reputation values of all nodes by aggregating both true and false LTOs, without any prior knowledge of which LTOs are true or false.

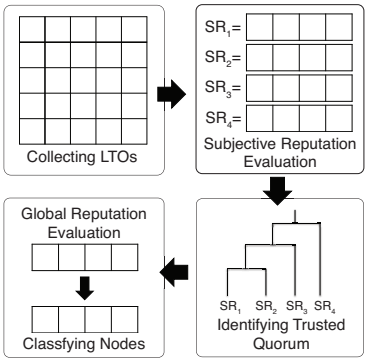


Fig. 1: Workflow of GlobalTrust

Assuming that honest nodes are a majority in the network, they are expected to form a consistent view on other nodes even in the presence of conflicting evidence. We call the view that node i has towards node j a *subjective reputation* (SR); it is computed by TA based on both the $LTO_{i,j}$ and the LTOs that other nodes have over node j. Section IV-B will detail how to compute the subjective reputation. We use a machine learning technique, called *hierarchical clustering*, to identify a minimum dominating set of nodes as a trusted quorum based on the similarity among their subjective views. Then we evaluate the reputation of a node by converging the subjective reputations of nodes in the quorum. Based on the computed reputation value of each node, TA judges the trustworthiness status of each node according to three reputation statuses:

honest, *malicious* or *unknown*. Fig. 1 summarizes the key processes of the *GlobalTrust*.

B. Subjective Reputation Evaluation

In our model, each node can compute LTOs only for other nodes it has directly interacted with, not for remote nodes because no direct evidence is available. However, TA can evaluate all nodes based on the LTOs provided by all nodes in the network. With the LTO matrix, TA will first calculate the subjective reputations (SR) of each node based on subjective trust values, LTOs, provided by all nodes in the network. Let $SR_{w,\,u}$ denote the reputation of node u evaluated by TA as it could have been subjectively assessed by node w. In the evaluation, node w trusts its own LTO. We use a weighted average to compute $SR_{w,\,u}$:

$$SR_{w,\,u} = \sum_{j \in S_u} LTO_{j,\,u} \cdot \frac{HR_j \cdot Sim(w,\,j)}{\sum_{j \in S_u} HR_j \cdot Sim(w,\,j)} \tag{4}$$

where S_u is the set of nodes that have non-$null$ LTOs over node u (including w if w has one), $LTO_{j,\,u}$ is the LTO of node j over node u, HR_j is node j's hierarchical rank, and $Sim(w,\,j)$ is the similarity between LTOs reported by node w and node j. The rationale behind the formula is as follows. From node w's viewpoint, to evaluate the reputation of another node u, besides its own direct observation (if any), the LTOs over node u reported by other nodes can be taken into consideration too. Node w weighs other nodes' $LTO_{j,\,u}$ values based on the similarity between its own view and node j's view. That is, it weighs more the opinions from others with more similar views to its own. The similarity of LTOs between node w and node j is measured based on a *cosine* function with the input of their LTO vectors:

$$Sim(w,\,j) = max(\cos(\mathbf{LTO'}_w,\,\mathbf{LTO'}_j),\,0). \tag{5}$$

Here we adopt a *cosine* function to capture the similarity of two LTOs represented by two vectors. The *cosine* similarity result is ranged in $[-1,\,1]$, where -1 refers to complete dissimilarity in the two opinions, 1 complete similarity, and 0 ignorance (uncertainty), indicating orthogonal opinions. Before computing the *cosine* similarity of two vectors, the LTOs in both vectors are linearly mapped to the scale of $[-1,\,1]$, re-scaled from the original scale in $[0,\,1]$. The re-scaled $\mathbf{LTO}$ vector is denoted as $\mathbf{LTO'}$. Note that if there is no common set between two vectors, the *cosine* similarity value is set to 0. Further, the similarity result is adjusted to 0 if the *cosine* similarity value of the two vectors is negative, which excludes evidence provided by untrusted nodes due to the dissimilarity. $SR_{w,\,u}$ is evaluated by:

$$SR_{w,u} = \begin{cases} \text{if } \sum_{j \in S_u} HR_j \cdot Sim(w,\,j) \neq 0, \\ \sum_{j \in S_u} LTO_{j,\,u} \cdot \frac{HR_j \cdot Sim(w,\,j)}{\sum_{j \in S_u} HR_j \cdot Sim(w,\,j)} \\ \text{else if } S_u \neq \varnothing, \\ \sum_{j \in S_u} LTO_{j,u} \cdot \frac{HR_j}{\sum_{j \in S_u} HR_j} \\ \text{else} \\ null \end{cases} \tag{6}$$

When $\sum_{j \in S_u} HR_j \cdot Sim(w,\,j) = 0$ (i.e., the denominator in Equation 4), this indicates that node w does not have directly observed evidence towards u, nor did any other nodes with whom node w shares positive similarity. In this case, we average the existing LTOs on node u with the HR of each recommender as the weight for $SR_{w,\,u}$, if any. If none of the nodes in the network has LTOs on node u (i.e., $S_u = \varnothing$), we set it to a $null$. Note that if $S_u = \varnothing$, $SR_{w,\,u}$ is $null$ for any w.

C. Assessment of Trusted Quorum

After computing the SR for each pair of nodes, TA generates a SR matrix. The SR tuple in node w's view is denoted as vector $\mathbf{SR}_w = (SR_{w,1}, \cdots, SR_{w,N})$. There are N SR tuples in total. Our next step is to identify a subset of the SR tuples as TA's *trusted quorum*. Intuitively, SR tuples from honest nodes tend to be similar and hence form a cluster, while those from malicious nodes may form another cluster or are irregularly distributed subject to specific false recommendation attack patterns. We call a cluster *dominating* if the number of nodes in the cluster exceeds the half of a network size. We aim to find the *minimum dominating cluster* to represent the trusted quorum. The reasons are two folds: the dominating size guarantees that the SR tuples in malicious nodes' views cannot form such a big cluster while the minimum requirement contributes to excluding inaccurate SR tuples, due to false reported LTOs and imperfect direct observations, as much as possible. We use the *agglomerative hierarchical clustering* technique to build a hierarchy of clusters based on all the SRs and find a minimum dominating cluster.

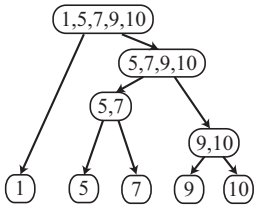


Fig. 2: An example of hierarchical clustering dendrogram

Fig. 2 is a simple example of *hierarchical clustering dendrogram*. In this method, each node starts with its own cluster, and the pairs of clusters with the nearest distance are merged continuously until only one cluster remains. Eventually it forms a hierarchical clustering tree. Here, the distance of two values a and b, denoted as $dist(a, b)$, is $|a - b|$, and the distance of two clusters A and B is defined as $\max\{dist(a, b) : a \in A, b \in B\}$. Fig. 2 describes the example procedures of hierarchical clustering denrogram as follows: (1) the cluster $\{9\}$ and the cluster $\{10\}$ are merged since their distance of 1 is the smallest; (2) the cluster $\{5\}$ and the cluster $\{7\}$ are merged because the current smallest distance is 2; (3) the cluster $\{5, 7\}$ and the cluster $\{9, 10\}$ are combined since the smallest distance becomes 5 after that; (4) the cluster $\{5, 7, 9, 10\}$ merges with the cluster $\{1\}$ to complete the hierarchical clustering.

Applying this method, we categorize N SR tuples into a hierarchical clustering tree by assigning each SR tuple into a leaf node. In our case, the distance between any two SR

tuples is their *Euclidean distance* and the distance between two clusters follows the same definition above. Therefore, the minimum dominating cluster, denoted as D, is the first cluster formed in the *agglomerative clustering* whose size is over $N/2$. This cluster D becomes TA's trusted quorum. To compute the *agglomerative clustering*, we use the *nearest-neighbor chain algorithm* [22]. The overall time and space complexity for the nearest nearest-neighbor chain algorithm is $O(N^2)$ and $O(N)$, respectively, where N is the number of nodes in the network.

D. Global Reputation Evaluation

We compute the global reputation of each node considering two aspects of reputation: *behavioral reputation* (BR) and *credibility reputation* (CR). Node u's behavioral reputation, BR_u, reflecting how other nodes view node u's network behavior, is computed by averaging the SR tuples in D:

$$BR_u = \begin{cases} \text{unknown} & \text{if } S_u = \varnothing \\ \frac{\sum_{w \in D} SR_{w,u}}{|D|} & \text{otherwise} \end{cases} \quad (7)$$

S_u is the set of nodes that have LTOs over node u, $SR_{w,u}$ is the SR of node u in node w's opinion. When no LTOs towards node u are available in the network (i.e., $S_u = \varnothing$), BR_u is set to $unknown$. In the case, $SR_{w,u}$ must be $null$ for any w, as mentioned previously.

Node u's credibility reputation, denoted as CR_u, indicates how trustworthy u's reported LTOs (i.e., $\mathbf{LTO}_u$) are. It is estimated based on the difference between u's reported LTOs and BRs of the nodes that node u has reported LTOs over. This implies that if the behavioral reputation of a node j is evaluated to be good, node u also has a very positive LTO over j, meaning u's LTO is more credible. The credibility of node u's LTOS, CR_u, is estimated by:

$$CR_u = \begin{cases} \text{unknown} & \text{if } \mathbf{LTO}_u = \text{null} \\ 1 - \sqrt{\frac{\sum_{j \in \{LTO_{u,j} \neq null\}} (LTO_{u,j} - BR_j)^2}{|\{j | LTO_{u,j} \neq null\}|}} & \text{otherwise} \end{cases} \quad (8)$$

Note that when node u does not report any LTOs (i.e., $\mathbf{LTO}_u = \text{null}$), $unknown$ is assigned to CR_u. In this case, its global reputation is solely computed based on its behavior.

Finally, TA computes the global reputation of node u by:

$$GR_u = \begin{cases} \gamma BR_u + (1-\gamma) CR_u & \text{if both } \textit{known} \\ \text{unknown} & \text{if both } \textit{unknown} \\ CR_u & \text{if only } BR_u = \textit{unknown} \\ BR_u & \text{if only } CR_u = \textit{unknown} \end{cases} \quad (9)$$

Here $\gamma \in [0,1]$ is used to normalize the global reputation values.

After TA computes global reputation (GR) values of all nodes, it can judge the trustworthiness of each node u as one of three statuses: $malicious$, $honest$, or $unknown$ by:

$$Decision(u) = \begin{cases} \text{unknown} & \text{if } GR_u = \textit{unknown} \\ \text{honest} & \text{if } GR_u \geq \theta \\ \text{malicious} & \text{if } GR_u < \theta \end{cases} \quad (10)$$

where θ is a decision threshold selected from the range in $[0,1]$ that may be adjusted to minimize detection errors (we will examine the impact of θ in our simulation experiments).

E. Security Analysis

For security analysis, let us first consider the case that a malicious node behaves consistently to other nodes (i.e., NMA, CRA, NRA and MSA attacks). In this case, honest nodes have high consistent views (LTOs) on every malicious node as well as on every honest node, meaning high similarity of LTOs between two honest nodes. On the other hand, the similarity of LTOs between an honest node and a malicious node depends on how faithfully the malicious node reported its LTOs. The more faithfully, the higher the similarity. Therefore, by converging the LTOs with their similarity as weight, $SR_{w,u}$ is highly accurate to reflect node u's behavioral reputation when node w is honest. That is, SR tuples in honest nodes' views are highly consistent and accurate. Note that for a malicious node w, $SR_{w,u}$ could be inaccurate if node w reports false LTOs, or accurate if $node$ reports LTOs honestly to actually contribute to reputation aggregation. By leveraging hierarchical clustering, the consistent and accurate SR tuples with a minimum dominating size will form a trusted quorum to eventually evaluate the behavioral reputations of all nodes accurately, which can effectively identify malicious nodes in the attacks including NMA, CRA and NRA. With the help of accurate behavioral reputation, the scheme can accurately evaluate the credibility reputation of nodes and hence effectively identify malicious spies in MSA.

There is a case that malicious nodes behave inconsistently to different honest nodes (i.e., CBA). Even in this case, since honest nodes have high consistent views on honest nodes consisting of a majority of the nodes in the network, they are more likely to form the trusted quorum even if malicious nodes may exhibits inconsistent network / reporting behavior. Note that if malicious nodes report their honest LTOs, they are likely to be involved into the trusted quorum and contribute to accurate reputation assessment. This, thus, enables their credibility reputations (CR) to maintain high. However, their behavioral reputations (BRs) will be low, which causes their overall global reputations lower than those of honest nodes. In this sense, our scheme is resilient against malicious nodes performing CBA and accordingly can effectively identify honest and malicious nodes, except the case with the following two cases: (1) when the ratio of malicious nodes is very close to 50% (see Section V-C for the analysis); and (2) the LTO matrix is too sparse, leading to the case the LTOs of malicious nodes form the majority in the LTO matrix.

V. Performance Evaluation

A. Simulation Setup

We evaluate *GlobalTrust* through extensive simulations using C. The network model uses a set of human-mobility traces from CRAWDAD [23]. In collection of the datasets, all participants were equipped with Global Positioning System (GPS) receivers to log their positions per 30 seconds. We use the dataset by KAIST (Korea Advanced Institute of Science and Technology) which uses mobility traces of 92 nodes. The nodes of a simulated network is split into two types of nodes,

Model	Behavior	Recommendation
NMA	misbehaving with prob. α	honestly reporting LTOs
NRA	misbehaving with prob. α	reporting opposite LTOs, $1-\alpha$
CRA	misbehaving with prob. α	reporting LTOs of 1 to malicious nodes and LTOs of 0 to honest nodes
MSA	half malicious nodes misbehaving with prob. α; the other half malicious nodes behaving honestly	reporting LTOs of 1 to malicious nodes and LTOs of 0 to honest nodes
CBA	misbehaving with prob. α to half honest nodes; behaving honestly to the other half honest nodes	reporting LTOs of 1 to malicious nodes and LTOs of 0 to honest nodes

TABLE I: Malicious attack patterns

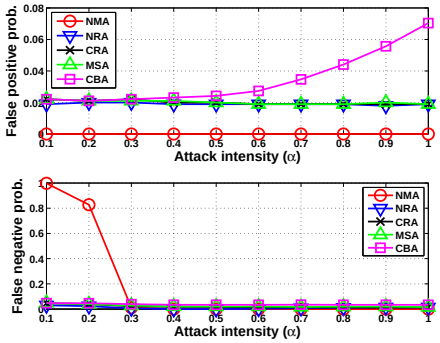


Fig. 3: Decision error vs. α

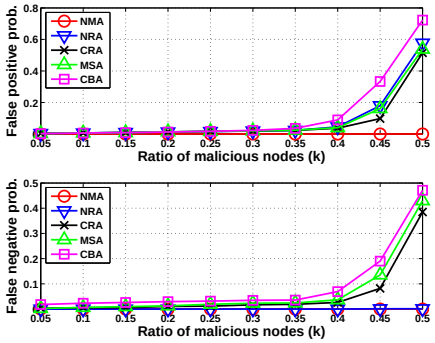


Fig. 4: Decision error vs. k

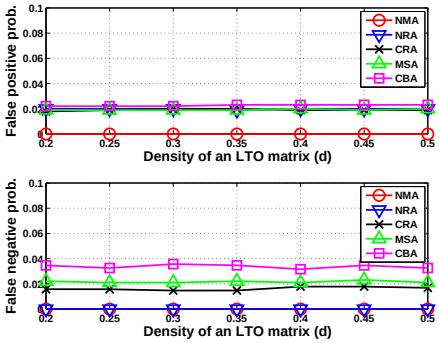


Fig. 5: Decision error vs. d

honest or malicious nodes. The numbers of malicious and honest nodes are denoted as m and h, respectively. The ratio of malicious nodes is denoted as k ($= \frac{m}{m+h}$) and its default value is set to 0.3. Each node is assumed to have an equal hierarchical rank, except TA, taking the role of a higher rank commander.

The networking traffic is simulated based on packet forwarding behavior. Every node randomly requests one of its neighboring nodes to forward a packet as a relay for 100 times per minute, where the one-hop wireless radio range is 250 meters. For honest nodes, they cooperate in forwarding packets with probability $(1-e) = 0.95$ and drop packets with probability $e = 0.05$. Honest nodes are supposed to provide LTOs of other nodes based on their direct observations. After a node forwards a packet to a neighbor node, it would monitor the neighbor's behavior on packet forwarding. Packet forwarding is regarded as positive behavior while packet dropping is counted as negative behavior. We consider the inherent detection error probability in the monitoring mechanism with ε = 0.05, providing falsely observed report towards the observed events (e.g., reporting opposite results). We summarize attackers' behavior pattern discussed in Section III-C in Table I. Note that α is the probability that a malicious node drops a packet and $\alpha = 0.5$ as the default.

TA computes the reputation of each node every 30 minutes. Based on TA's online interval and mobility traces, we observe that on average a node encounters with 39% of all nodes as a 1-hop neighbor. The LTOs submitted in the previous offline time frame (i.e., the last 30 minutes) are collected to estimate global reputations and make decisions about nodes' statuses (i.e., honest, malicious, or unknown). The coefficient γ is set to 0.7 to weigh the behavioral reputation (BR) higher than the credibility reputation (CR) because malicious behavior is able to cause direct attacks to the network performance (e.g., throughput), whereas false LTOs may be filtered out by GlobalTrust and hence introduce less negative impacts on the network. We set the decision threshold, $\theta = 0.8$, to determine whether a node is malicious or honest. The simulation is run 1000 for each scenario for the results shown here.

B. Performance Metrics

		TA Decision	
		Malicious	Honest
Ground Truth	Malicious	true positive (TP)	false negative (FN)
	Honest	false positive (FP)	true negative (TN)

TABLE II: Detection types

In this work, we consider detection errors (i.e., FPs and FNs) on trust decisions evaluated by TA as performance metrics. We show all possible decision cases in Table II. For the nodes classified as *unknown*, this is the case when they neither provide any recommendation nor interact with any other nodes, regarded as inactive in the network. We do not consider them for our performance analysis. For an active node, four outcomes are possible, as in Table II. We mainly use both false positive (FP) and false negative (FN) probabilities as our performance metrics to indicate judgment (decision) errors. Besides, we use receiver operating characteristics (ROC) analysis as a performance metric, indicating correct detection probability.

C. Comprehensive Evaluation

This subsection gives a comprehensive evaluation of decision errors with respect to three factors: probability of attack intensity (α), ratio of malicious nodes (k), and malicious attack patterns. Finally, we show how the selection of decision threshold (θ) affects the decision accuracy.

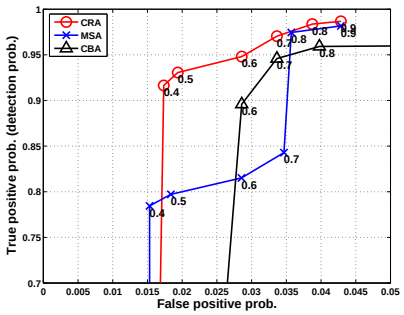


Fig. 6: ROC curve by varying θ

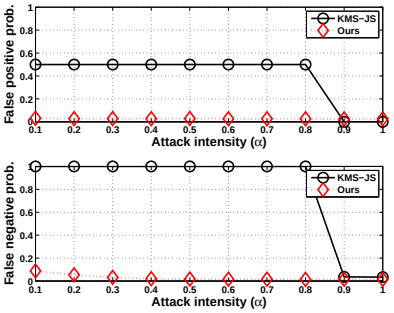
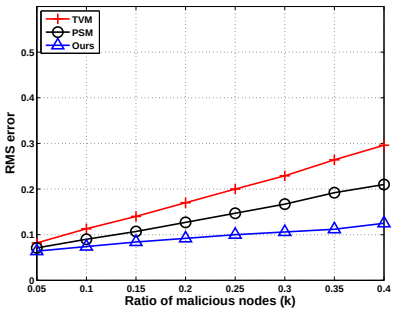
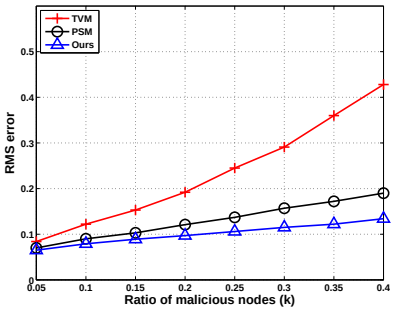


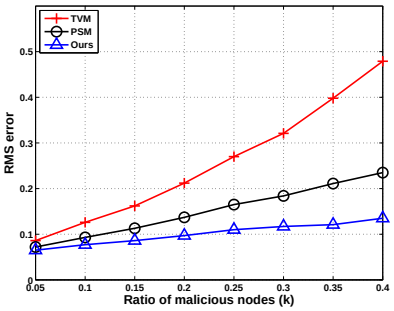
Fig. 7: Comparison with KMS-JS



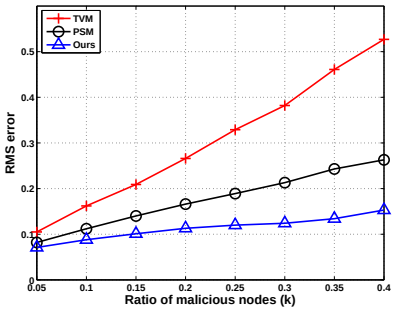
(a) RMS error vs. k under NRA



(b) RMS error vs. k under CRA



(c) RMS error vs. k under MSA



(d) RMS error vs. k under CBA

Fig. 8: Comparison with PeerTrust in accuracy

Fig. 3 illustrates how the decision errors vary as attack intensity, α, increases. We observe that the maximum FP is less than 0.03 over the entire range of α except for CBA. Under NMA, the FP is close to zero because of no false (dishonest) recommendations. Under CBA, the FP increases slightly up to 0.07 when α increases. This is because the increasing α can increase the dissimilarity between honest nodes' LTOs and malicious nodes' LTOs, which ultimately affects the subjective reputation evaluation. However, the negative impact is small because the similarity of honest nodes' LTOs over honest nodes ensures the credibility of LTOs.

Fig. 3 also shows that the observed FN is close to 0, except the case of NMA with $\alpha < 0.3$ showing a significant number of malicious nodes is falsely identified as honest. With small α, malicious nodes under NMA do not exhibit misbehavior much, accordingly leading to high FN. Under all other attacks, malicious nodes' global reputations are downgraded due to their misbehavior and false LTOs, leading to the situation that most of them are classified as malicious even with low α. Besides the spies in MSA are well identified even if they show consistent honest behaviors. Therefore, considering credibility reputation (CR) in deriving the overall global reputation significantly helps identify malicious nodes showing inconsistent behavior such as intermittent reporting of false recommendations.

Fig. 4 reveals how the increasing ratio of malicious nodes, k, degrades the decision errors. The FP increases but stays below 0.1 under all types of attacks with $k < 0.4$. When k increases up to 0.5, the FP increases rapidly and stays around 0.7 at the end. This is because the trusted quorum derived from the hierarchical clustering may include more malicious nodes than honest nodes when malicious nodes become a majority of the network. We also observe that CBA is more detrimental than NRA, CRA and MSA when k increases. For attacks such as NRA, CRA and MSA, a higher k means more malicious nodes in the trusted quorum. However, when malicious nodes perform the CBA attack, a higher k not only increases the degree of malicious activities, but also affects the subjective reputation evaluation by honest nodes, because the CBA attackers generate more conflicting LTOs.

When k is below 0.4, the FN increases with k under MSA, CRA and CBA and it reaches 0.06 at its maximum under CBA whereas it is almost zero under NMA and NRA. For higher $k > 0.4$, the FN increases as quickly as the FP increases because the trusted quorum tends to include more malicious nodes. Again, CBA impacts more detrimentally than CRA and MSA over a wider range of k. This is because malicious nodes performing the CBA can obtain high trust values in behavior reputation (BR) while still performing attacks, compared to malicious nodes performing CRA and MSA. Overall GlobalTrust is fairly resilient to the attacks considered above when $k < 0.4$.

Fig. 5 shows how the density of an LTO matrix, d, affects the decision errors. The result shows that both FP and FN stay low and they fluctuate a little bit as the density of an LTO matrix, d, increases under all other attacks considered. Hence, *GlobalTrust* is adaptive to a wide range of LTO density, d, as low as 20%.

Fig. 6 visualizes how *GlobalTrust* performs w.r.t. θ using the ROC metric. We consider three collusion attacks including CRA, MSA and CBA. The y-axis is the TP probability, referring to the probability of correctly detecting malicious nodes, while the x-axis denotes the FP, meaning the probability of detecting a good node as bad. The value labeled with each point is the decision threshold, θ. The observed general trend is that the TP probability increases with θ and FP (< 0.05). In Fig. 6, to ensure that ROC (detection probability) is above 0.7, θ should be as low as 0.4, 0.4 and 0.6 under CRA, MSA and CBA, respectively. Similar to our observation in previous results, malicious nodes performing CBA have higher reputation values than those performing CRA and MSA.

When the threshold, θ, increases from 0.7 to 0.8 under MSA, ROC significantly increases by 0.13 between these two thresholds. This implies that sufficiently high θ is required to maximize ROC. We observe that $\theta = 0.8$ is optimal under the given condition because this ensures the smallest fluctuation of FP and FN, 0.05, under the considered attacks.

D. Comparative Performance Analysis

This subsection presents two performance comparison studies: (1) *GlobalTrust* vs. *k-means clustering-based judgment* scheme [17]; and (2) *GlobalTrust* vs. two existing reputation methods (i.e., TVM and PSM) in *PeerTrust* [3].

1) GlobalTrust vs. K-Means Clustering-Based Judgment: Reidt *et al.* [17] introduced a *k-means clustering-based judgment scheme* (KMS-JS) on a trust overlay network. In KMS-JS, TA collects all LTOs to form LTO matrix O, in which $o_{i,j}$ represents the LTO of node i about node j. All $N \times N$ entries are assumed to be full after a sufficiently long time elapsed, where N is the number of nodes in the network. The LTOs over node j are placed in its column vector of the matrix O, $\mathbf{o_j} = (o_{1,j}, \ldots, o_{N,j})$. The values in column vectors of honest nodes tend to be close to each other and thus can often be clustered together. The judgment system uses a $N-1$ dimensional hyper-plane to maximally separate two clusters based on nodes' column vectors, and the larger cluster is categorized as honest. Unfortunately, the decision made may not be true, showing severe security vulnerability due to conflicting recommendation attacks. For example, a collusive community divides all honest nodes (i.e., nodes out of the community) into two groups equally, denoted as $\mathcal{G}_1$ and $\mathcal{G}_2$; collusive malicious nodes provide highest LTOs about themselves and honest nodes in $\mathcal{G}_1$, while they provide lowest LTOs about honest nodes in $\mathcal{G}_2$; also, malicious nodes control their attack intensity α in a proper level. This attack pattern tends to maximize the difference between vectors $\mathbf{o_j}$ of two different honest groups and minimize the difference between malicious nodes and nodes in $\mathcal{G}_1$. Under this attack, the judgment system may cluster malicious nodes and nodes $\mathcal{G}_1$ into the honest class while nodes in $\mathcal{G}_2$ are clustered into the malicious class. Fig. 7 shows how detection error (FP and FN) varies with respect to α, when the ratio of malicious nodes k is 0.3. Fig. 7 shows that KMS-JS performs very poorly with FN close to 1 and FP close to 0.5 for $\alpha < 0.9$. In contrast, *GlobalTrust* performs significantly better than KMS-JS, with both FN and FP less than 0.1 in most cases when $\alpha > 0.1$.

TBRM	Cons.	NMA	NRA	CRA	MSA	CBA
CORE [24]	Yes	$\surd$	$\surd$	$\bullet$	$\bullet$	$\bullet$
EigenTrust [7]	Yes	$\surd$	$\surd$	$\star$	$\bullet$	$\star$
SORI [12]	No	$\surd$	$\surd$	$\bullet$	$\bullet$	$\bullet$
Robust [11]	No	$\surd$	$\surd$	$\surd$	$\star$	$\bullet$
PSM [3]	No	$\surd$	$\surd$	$\surd$	$\star$	$\surd$
PowerTrust [8]	Yes	$\surd$	$\surd$	$\surd$	$\star$	$\surd$
GlobalTrust	Yes	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
$\surd$: resilient; $\star$: partially vulnerable; $\bullet$: vulnerable						

TABLE III: Comparison between our GlobalTrust and existing TBRM schemes w.r.t. consistency and resilience

2) GlobalTrust vs. Existing Reputation Schemes: Here we compare *GlobalTrust* with the existing reputation schemes [3], [7], [8], [11], [12], [24] based on two criteria: consistency and resilience, shown in Table III. In Fig. 8, we compare *GlobalTrust* with two reputation techniques used in *PeerTrust* [3], trust value based credibility measure (TVM) and personalized similarity measure (PSM), with respect to the accuracy of trust assessment.

Consistency: For fully distributed tactical networks, reputation evaluation is normally performed either in a distributed, cooperative way [7] or in an independent, uncooperative way [3]. In the former case, the evaluated reputation of a node must be consistent through the network. In the latter case, the evaluated reputation towards a node may be inconsistent in the network if an evaluator differentiates direct observations from indirect observations in deriving reputation values. Table III shows if existing reputation schemes have considered view consistency.

Resilience: We compare *GlobalTrust* with existing reputation schemes w.r.t. their resilience to the types of attacks in Table III. *CORE* and *SORI* do not deal with collusion attacks such as CRA, MSA and CBA. *EigenTrust* is able to resist CRA to some extent with the help of pre-trusted nodes; however, for those nodes that the pre-trusted nodes have not had a chance to interact or observe (i.e., high uncertainty), the reputation evaluation would be highly distorted. Besides, MSA is an attack that can effectively defeat *EigenTrust* based on two reasons: (1) *EigenTrust* has no way to identify spies since their reputations are overestimated with high reputation values; and (2) false recommendations provided by spies are regarded as trustworthy information because the spy nodes do not show other abnormal behavior except passing false recommendations. *EigenTrust* may be vulnerable to CBA when pre-trusted nodes may be cheated by malicious nodes showing inconsistent behavior. *Robust*, *PSM* and *PowerTrust* devise trust models to effectively filter out false recommendations by collusion attacks; however, they do not consider credibility of recommendations for reputation evaluation and hence cannot identify spies in MSA. *Robust* is vulnerable to CBA with a malicious node showing inconsistent behavior because of the lack of capability to detect them by honest nodes. Since *GlobalTrust* can filter out false recommendations using the subjective reputation of nodes based on the identified trust quorum. In addition, *GlobalTrust* uses credibility reputation

(CR) to consider credible recommendations that can help correctly measure global reputation, ultimately leading to effectively identifying spies in MSA.

Accuracy: In Fig. 8, we compare *GlobalTrust* with *PeerTrust* [3] w.r.t. *accuracy of trust assessment*. We choose *PeerTrust* for the comparison because *PeerTrust* and *GlobalTrust* adopt the same definition of behavior reputation. The two evaluation models, TVM and PSM, in *PeerTrust* are devised based on different strategies to estimate recommendation credibility. TVM is known as vulnerable to collusion attacks while PSM is well designed to resist CRA. All parameters are set equally for these schemes in our simulation for fair comparison, as shown in Section V.

For PSM model, a honest node is randomly assigned as the evaluator to compute reputation-based trust values of all nodes. The evaluator's LTOs are *pre-trusted* in PSM when estimating the credibility of others' recommendations. We compare these three reputation evaluation methods (TVM, PSM and *GlobalTrust*) w.r.t. judgment accuracy under NRA, CRA, MSA and CBA. We use the root-mean-square (RMS) of the behavioral reputations of all nodes and the *actual* likelihood that all nodes behave honestly to measure reputation evaluation errors. That is, we compare the behavioral reputations (BRs) in *GlobalTrust* with the reputation-based trust values in TVM and PSM since all of these values estimate the actual probability that all nodes behave honestly.

The actual behavioral reputation towards a malicious node's behavior is $1-\alpha$ under NRA and CRA, 1 for spy and $1-\alpha$ for non-spy under MSA, and $1-\frac{\alpha}{2}$ under CBA. The actual reputation of an honest node's behavior is 1. Fig. 8 shows the results comparing *GlobalTrust*, TVM and PSM. We mainly observe the following trends:(1) TVM is severely vulnerable to collusion attacks including CRA, MSA and CBA as the RMS error has exceeded 0.4 when the ratio of malicious nodes, k, reaches 0.4; (2) *GlobalTrust* has about 0.25 to 0.4 lower RMS evaluation errors than PSM when k reaches 0.4 for each attack; and (3) PSM performs well, being resilient against NRA and CRA since the increased span of the RMS error is not significantly large (around 0.1) when k varies from 0.05 to 0.4. In contrast to PSM, *GlobalTrust* performs well in interpreting the behavioral reputation of a node under all these attacks as the maximum RMS error increases approximately up to 0.05. The results prove that emphGlobalTrust outperforms TVM and PSM in terms of the accuracy of trust assessment.

VI. Conclusion

In this paper, we proposed a trust-based reputation scheme, called *GlobalTrust*, to accurately evaluate the reputation of nodes reflecting both the behavioral trustworthiness and recommendation credibility in a tactical network environment, where malicious entities exists while no pre-trusted nodes are assumed except a commander node. Through extensive simulation experiments, we compared *GlobalTrust* with other existing schemes and showed that *GlobalTrust* outperforms existing counterparts in terms of being highly resilient against various types of attacks, maintaining high view consistency throughout the network, and generating low reputation judgment errors.

Acknowledgement: The work of Sencun Zhu was supported in part by NSF grant CCF-1320605 and a Google gift. We also thank the reviewers for helpful comments.

References

[1] S. Reidt and S. D. Wolthusen, "Efficient distribution of trust authority functions in tactical networks," in *Information Assurance and Security Workshop*, 2007.

[2] J. Cho, A. Swami, and I. Chen, "A survey of trust management in mobile ad hoc networks," *IEEE Communications Surveys and Tutorials*, vol. 13, no. 4, pp. 562–583, 2011.

[3] L. Xiong and L. Liu, "Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843–857, 2004.

[4] H. Li and M. Singhal, "Trust management in distributed systems," *Computer*, vol. 40, no. 2, pp. 45 –53, Feb. 2007.

[5] K. Govindan and P. Mohapatra, "Trust computations and trust dynamics in mobile adhoc networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 2, pp. 279–298, 2012.

[6] K. Aberer and Z. Despotovic, "Managing trust in a peer-2-peer information system," in *Proc. CIKM*, 2001.

[7] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proc. WWW*, 2003.

[8] R. Zhou and K. Hwang, "Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 4, pp. 460–473, Apr. 2007.

[9] G. Bella, G. Costantino, and S. Riccobene, "Managing reputation over manets," in *Information Assurance and Security*, 2008.

[10] G. Arboit, C. Crépeau, C. R. Davis, and M. Maheswaran, "A localized certificate revocation scheme for mobile ad hoc networks," *Ad Hoc Netw.*, vol. 6, no. 1, pp. 17–31, Jan. 2008.

[11] S. Buchegger and J. Le Boudec, "A robust reputation system for mobile ad-hoc networks," in *Proc. P2PEcon*, 2003.

[12] Q. He, D. Wu, and P. Khosla, "Sori: a secure and objective reputation-based incentive scheme for ad-hoc networks," in *Proc. WCNC*, 2004.

[13] W. L. Teacy, J. Patel, N. R. Jennings, and M. Luck, "Travos: Trust and reputation in the context of inaccurate information sources," *Autonomous Agents and Multi-Agent Systems*, vol. 12, no. 2, pp. 183–198, 2006.

[14] A. Jsang and R. Ismail, "The beta reputation system," in *Proceedings of the 15th bled electronic commerce conference*, 2002.

[15] H. Chan, V. D. Gligor, A. Perrig, and G. Muralidharan, "On the distribution and revocation of cryptographic keys in sensor networks," *IEEE Trans. Dependable Secur. Comput.*, vol. 2, no. 3, pp. 233–247, Jul. 2005.

[16] M. Raya, M. H. Manshaei, M. Félegyhazi, and J.-P. Hubaux, "Revocation games in ephemeral networks," in *Proc. CCS*, 2008.

[17] S. Reidt, M. Srivatsa, and S. Balfe, "The fable of the bees: incentivizing robust revocation decision making in ad hoc networks," in *Proc. CCS*, 2009.

[18] X. Chen, H. Patankar, S. Zhu, M. Srivatsa, and J. Opper, "Zigzag: Partial mutual revocation based trust management in tactical ad hoc networks," in *Proc. SECON*, 2013.

[19] S. Buchegger and J. Le Boudec, "Performance analysis of the confidant protocol," in *Proc. MobiHoc*, 2002.

[20] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing Robust and Ubiquitous Security Support for Mobile Ad-hoc Networks," in *IEEE ICNP*, 2001.

[21] F. Mármol and G. Pérez, "Security threats scenarios in trust and reputation models for distributed systems," *computers & security*, vol. 28, no. 7, pp. 545–556, 2009.

[22] J. Benzécri, "Construction d'une classification ascendante hiérarchique par la recherche en chaîne des voisins réciproques," *Les Cahiers de l'Analyse des Données*, vol. 7, no. 2, pp. 209–218, 1982.

[23] I. Rhee, M. Shin, S. Hong, K. Lee, S. Kim, and S. Chong, "Crawdad, data set ncsu/mobilitymodels (v. 2009-07-23)," July 2009.

[24] P. Michiardi and R. Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Advanced Communications and Multimedia Security*, 2002.